Machine Learning Based Road Damage Detection and Geo Tagging System Using Embedded Edge Computing

***Abstract*:** Road infrastructure monitoring is traditionally carried out through manual inspection, which is time consuming, error-prone, and inefficient for large-scale use. This project proposes an embedded, real-time road damage detection system that combines computer vision, GPS-based geo-tagging, and edge computing on a Raspberry Pi platform. A YOLO based model detects potholes, cracks, and manhole anomalies from live footage, recording timestamps and location data for each detection. A Road Health Score (RHS) is then calculated to assess road condition and support maintenance prioritization. The system offers a low-cost, scalable solution for smart-city applications.

***Keywords*:** Road Damage Detection, YOLO, Edge AI, Raspberry Pi, Smart Infrastructure, Computer Vision, Geo-Tagging

I. Introduction

Efficient road maintenance is essential for transportation safety and economic development. Traditional inspection methods rely on manual surveys, which are labour intensive and lack scalability. With advances in embedded systems and artificial intelligence, automated road monitoring can now be achieved using lightweight hardware and real-time object detection models.This project proposes a smart road inspection system capable of detecting surface defects and geo-tagging them automatically using edge computing.

II. System Architecture

The proposed system is an embedded edge-computing solution designed to detect and geo-tag road surface damages in real time. It integrates a Raspberry Pi, camera module, GPS receiver, and a YOLO-based deep learning model into a single processing pipeline. The camera continuously captures road images, which are analyzed locally on the Raspberry Pi without relying on cloud connectivity. When a defect such as a pothole, crack, or manhole is detected, the system retrieves GPS coordinates, records the timestamp, and stores the detection data. This architecture enables low-cost, portable, and scalable road monitoring suitable for smart transportation applications.

1.Continuous Video Acquisition: The camera module mounted on the vehicle continuously captures live video of the road surface. This ensures uninterrupted monitoring of road conditions during movement. The captured video stream is divided into individual frames, which are then forwarded to the processing unit for analysis.

2. Frame Processing on Raspberry Pi

Each captured frame is processed locally on the Raspberry Pi, which acts as the edge computing device. Performing computation on-device eliminates the need for cloud connectivity, reduces latency, and enables real-time analysis even in remote areas. The Raspberry Pi prepares the frames by resizing and formatting them to match the input requirements of the detection model.

3. Damage Detection Using YOLO Model

The pre-trained YOLO (You Only Look Once) deep learning model analyses each frame to identify road defects such as potholes, cracks, and damaged manholes. YOLO performs object detection in a single pass, allowing fast and efficient classification along with bounding box localization and confidence scoring.

4. GPS-Based Location Retrieval

When a defect is detected, the system communicates with the GPS module to obtain the precise geographic coordinates (latitude and longitude) of that location. This geo-tagging enables mapping of detected damages and helps authorities identify the exact repair site.

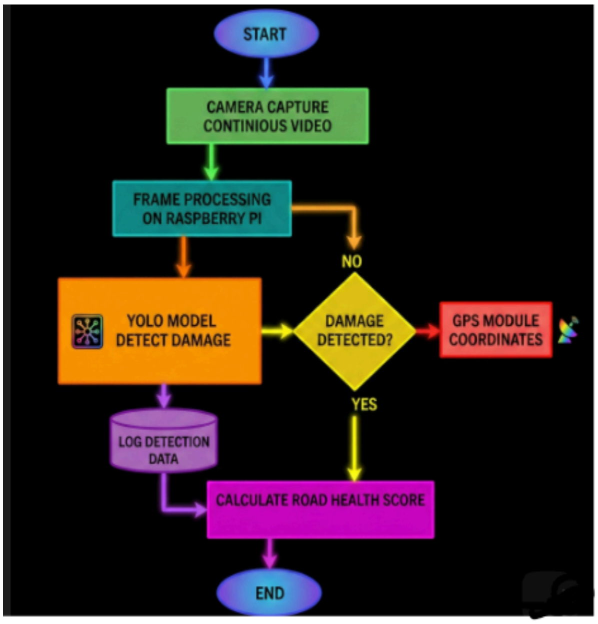
5. Detection Logging and Evidence Storage

After validation, the system records the detection details, including:

- Type of damage detected
- Timestamp of detection
- GPS coordinates
- Captured image of the defect

6. Road Health Score Computation

The collected detection data is analysed to compute a Road Health Score (RHS), which quantitatively represents the condition of the monitored road segment. The score is calculated based on the number and severity of detected defects, allowing roads to be categorized as good, moderate, poor, or critical. This scoring mechanism supports data-driven maintenance prioritization.



III. HardWare Components

COMPONENTS	FUNCTIONS
Raspberry Pi	Edge Processing Unit
Camera Module	Captures real Time Footage
GPS Module	Provides Latitude & Longitude
SD Card	Stores Logs And Captured Images
Power Supply	Enables Poratble Operation

The proposed system is implemented using a compact embedded hardware setup designed for real-time processing and portability. Each component plays a specific role in enabling image acquisition, computation, localization, and data storage..

Raspberry Pi (Edge Processing Unit)

The Raspberry Pi serves as the central processing unit of the system and performs all computational tasks locally. It is responsible for running the trained YOLO deep learning model, processing incoming video frames, and coordinating communication between the camera and GPS module. By performing inference on-device, the Raspberry Pi eliminates the need for cloud-based processing, thereby reducing latency, improving response time, and allowing the system to function in areas with limited internet connectivity. Its low power consumption and compact size make it suitable for deployment in mobile environments such as vehicles.

Camera Module (Image Acquisition Unit)

The camera module is used to capture continuous real-time footage of the road surface. It provides the visual input required for the computer vision model to analyse road conditions. The camera is configured to operate at a resolution optimized for real-time inference, ensuring a balance between detection accuracy and computational efficiency. Mounted on the vehicle, it enables continuous monitoring of road conditions during motion

GPS Module (Localization Unit)

The GPS module provides geographic positioning information corresponding to each detected road defect. When the detection model identifies a pothole, crack, or manhole, the system retrieves the latitude and longitude from the GPS receiver. This geo-tagging capability allows defects to be mapped spatially, enabling maintenance teams to locate and repair damaged sections accurately. The GPS module communicates with the Raspberry Pi through serial communication.

SD Card (Data Storage Unit)

- The SD card acts as the primary storage medium for the system. It stores:
- Captured images of detected damages
- Detection logs and metadata
- Model files and system software

Power Supply (Portable Operation Unit)

A regulated power supply powers the Raspberry Pi and connected peripherals. The system is designed to operate using portable power sources such as battery packs, enabling deployment in field conditions without fixed infrastructure. Stable power delivery is essential to maintain uninterrupted processing and avoid system shutdown during real-time monitoring.

IV Machine Learning Methodology

Overview of AI-Based Road Damage Detection

The proposed system employs a deep learning-based object detection approach to automatically identify road surface defects in real-time. Traditional manual inspection and rule-based image processing methods are inefficient and inconsistent when applied to large-scale infrastructure monitoring. Therefore, an AI-based detection model is used to ensure accurate and continuous monitoring.

Key Features of AI-Based Detection:

- Automated defect identification
- Real-time processing capability
- Multi-class object detection
- High adaptability to different road conditions

Dataset Collection and Preparation
A diverse dataset is essential for training a robust detection model capable of performing in real-world environments.

Dataset Characteristics:
- Images of damaged and non-damaged roads
- Different lighting conditions (daylight, shadow, low light)
- Various road textures (asphalt, concrete)
- Urban and semi-urban road scenarios

The inclusion of diverse data improves the model's generalization ability and prevents overfitting.

Data Annotation Process
Data annotation is performed using bounding boxes to label the location and type of road damage within images.

Annotated Classes:
- Pothole
- Crack
- Manhole

Each image is manually labelled to ensure supervised learning efficiency. Proper annotation allows the model to learn spatial and visual patterns associated with road defects.

Model Selection YOLO Algorithm
The YOLO (You Only Look Once) object detection model is selected due to its real-time detection capability and lightweight architecture suitable for embedded systems.

Reasons for Choosing YOLO:
- Single-stage detection architecture
- Fast inference speed
- High accuracy for object localization
- Suitable for edge computing devices
- Ability to detect multiple objects in one frame

Unlike traditional CNN models that process regions separately, YOLO processes the entire image in a single pass, making it highly efficient for real-time road monitoring applications.

Training and Optimization Strategy
The model training process involves multiple optimization techniques to achieve stable and accurate detection.

Training Steps:
- Image preprocessing and resizing
- Dataset splitting (Train, Validation, Test)
- Data augmentation techniques
- Hyperparameter tuning
- Confidence threshold adjustment

Optimization Techniques Used:

- Reduced input resolution for faster inference
- Confidence threshold set to balance precision and recall
- CPU optimization for Raspberry Pi deployment

These strategies ensure that the model performs efficiently even on limited-resource embedded hardware.

V EMBEDDED SYSTEM IMPLEMENTATION

The proposed system is designed based on an edge computing architecture where the trained deep learning model is deployed directly on the Raspberry Pi device. Instead of sending captured images to a cloud server for processing, all computations including image acquisition, object detection, and logging are performed locally on the embedded system. This design choice significantly reduces system latency and enables real-time road damage monitoring even in areas with limited or no internet connectivity.

Deploying the model on an edge device ensures that the system operates autonomously and continuously without relying on external infrastructure. In practical road inspection scenarios, network availability cannot always be guaranteed, especially in rural or highway environments. Therefore, edge AI deployment enhances the reliability and robustness of the system.

Furthermore, performing inference locally reduces transmission delays and allows immediate detection of road damages such as potholes and cracks. This is particularly important for smart transportation and real-time maintenance applications where quick decision-making is required.

Software Architecture Integration

The software architecture of the system is modular and designed using Python due to its flexibility and extensive support for computer vision and embedded system libraries. The integration of multiple software components ensures smooth coordination between hardware modules and AI processing.

The system utilizes a multi-threaded approach where GPS data acquisition runs in parallel with the main detection loop. This improves efficiency and prevents delays in real-time frame processing. The architecture ensures that each module performs a specific task while maintaining synchronization with the overall pipeline.

Major Software Modules Used:

- OpenCV for real-time image processing and visualization
- Picamera2 for continuous camera frame capture
- YOLO framework for deep learning-based object detection
- Serial communication library for GPS data reading
- Threading module for asynchronous GPS tracking

Each module is interconnected to form a complete embedded AI system. The camera continuously streams frames, which are then processed by the YOLO model. Simultaneously, the GPS thread updates the location data, which is later appended to stored detections. This integrated architecture improves system efficiency and scalability.

Real-Time Processing Workflow

The system follows a structured and automated workflow to ensure uninterrupted monitoring of road conditions. The workflow is optimized for embedded hardware constraints while maintaining detection accuracy.

Initially, the camera captures live frames of the road surface in real time. These frames are preprocessed and passed to the YOLO detection model for inference. The model then identifies road defects and generates bounding boxes along with class labels.

Step by Step Execution Flow:

1. Capture live frame using camera module
2. Preprocess frame for model compatibility
3. Perform YOLO-based object detection
4. Draw bounding boxes and class labels
5. Assign severity priority to detected damages
6. Fetch GPS coordinates from the GPS module
7. Store image and metadata automatically
8. Display annotated output on the embedded screen

This continuous loop ensures that the system operates in real time and provides instant visual feedback along with automated data logging

VI Detection and Geo Tagging Pipeline

The visual output confirms that the system performs stable detection even under embedded processing constraints.

Real-Time Damage Detection Mechanism

The detection pipeline is the core component of the proposed system, responsible for identifying road damages from live video input. The trained YOLO model processes each frame and detects defects such as potholes and cracks with bounding box localization. The model operates at a predefined confidence threshold to balance detection accuracy and computational efficiency.

The prototype output clearly demonstrates that the system can detect multiple road damages simultaneously within a single frame. Bounding boxes are drawn around detected objects, and labels are displayed in real time, which validates the effectiveness of the embedded AI model.

Detection Capabilities:

- Real-time pothole detection
- Crack identification under varying textures
- Multi-object detection in a single frame
- Accurate bounding box visualization
- Continuous frame-by-frame monitoring

Geo-Tagging and Location Tracking

To enhance the practical usability of the system, a GPS module is integrated for geo-tagging detected road damages. The GPS continuously reads NMEA data through serial communication and extracts latitude and longitude coordinates. These coordinates are attached to each stored detection record, enabling location-based road condition mapping.

Geo-tagging plays a crucial role in smart infrastructure systems as it allows authorities to identify the exact location of road defects for maintenance planning. The system is capable of storing coordinates along with timestamps and damage class information

Functions of Geo-Tagging:

- Real-time location acquisition
- Mapping of road damage hotspots
- Location-based maintenance scheduling
- Infrastructure analytics and reporting

Automated Data Storage System

The system incorporates an automated evidence storage mechanism that logs every valid detection along with associated metadata. Whenever a new road damage is detected and validated by the duplicate filtering algorithm, the system captures and stores the frame as an image file.

Each stored record contains structured information that supports further analysis and documentation.

Stored Information Includes:

- Captured detection image
- Detected class label (pothole/crack)
- Priority level
- Timestamp of detection
- GPS location

This automated logging system ensures traceability and provides reliable evidence for infrastructure inspection reports and smart city databases.

VII Duplicate Detection Filtering

In real-time video analysis, the same road defect may appear in multiple consecutive frames due to continuous camera capture. Without an efficient filtering mechanism, the system would repeatedly store identical detections, leading to redundant data accumulation and inefficient storage usage.

Duplicate detections can negatively impact the reliability of analytics by inflating the number of detected damages. Therefore, an intelligent filtering mechanism is necessary to maintain data accuracy

Problems Without Duplicate Filtering:

- Excessive memory consumption
- Incorrect damage statistics
- Redundant image storage
- Reduced processing efficiency

Filtering Methodology:

- Spatial distance threshold between detections
- Time threshold for repeated occurrences
- Label consistency verification

This approach ensures that only unique and meaningful detections are stored, eliminating redundant or irrelevant data. By filtering out duplicate or insignificant records, the system reduces unnecessary processing and storage overhead. This streamlined handling of information directly enhances system efficiency, allowing faster analysis and better resource utilization. At the same time, it guarantees that road damage records remain accurate, reliable, and useful for long-term monitoring, planning, and maintenance.

VIII PRIORITY-BASED DAMAGE CLASSIFICATION

Severity-Based Classification Logic

The system assigns a priority level to each detected road damage based on its severity and potential impact on road safety. This classification enhances the decision-making process for maintenance authorities and infrastructure management systems.

Priority Mapping Strategy:

- Pothole → High Priority (Severe damage)
- Manhole → Medium Priority (Moderate risk)
- Crack → Low Priority (Minor damage)

Potholes are categorized as high priority because they pose a significant safety risk to vehicles and pedestrians. Cracks, although less severe, can gradually develop into major road failures if not monitored.

Importance of Priority-Based Monitoring

Priority-based classification transforms raw detection outputs into actionable insights for smart infrastructure systems. Instead of treating all damages equally, the system intelligently ranks defects based on urgency**.**

Key Benefits:

- **Faster maintenance response**
- **Efficient resource allocation**
- **Improved road safety management**
- **Support for predictive maintenance**
- **Smart city infrastructure integration**

This approach enables government agencies to focus on critical road damages first, reducing accident risks and long-term repair costs.

IX EXPERIMENTAL RESULTS AND PROTOTYPE OUTPUT ANALYSIS

Prototype Testing Description

The developed prototype was experimentally tested using a live camera feed integrated with a Raspberry Pi embedded system. The objective of testing was to evaluate real-time detection accuracy, storage functionality, and system stability under continuous operation.

The system operated in an indoor testing environment with real-time frame capture and AI inference enabled. Despite limited computational resources, the prototype successfully demonstrated continuous detection and logging of road damages.

Testing Conditions:

- Live camera input
- Embedded CPU-based inference
- Continuous detection loop
- Automated logging enabled
- Indoor experimental setup

Console Output Validation

The console output generated during testing confirms the successful functioning of the automated detection and storage system. Multiple detections of potholes and cracks were recorded with accurate timestamps and priority levels.

Observations from Prototype Console Logs:

- Repeated pothole detections with HIGH priority
- Crack detections with LOW priority
- Automatic image capture and saving
- Timestamp-based logging for each detection
- Stable and continuous system execution

This validates that the system is capable of performing real-time monitoring and documentation of road damages without manual intervention

Performance Evaluation

The overall system performance was found to be stable and reliable during prototype testing. The model successfully processed frames in real time and generated accurate bounding box detections for road defects.

Performance Highlights:

- Successful real-time detection on embedded device
- Accurate classification of potholes and cracks
- Efficient image storage mechanism
- Smooth visual output display
- Stable CPU-based inference execution

The embedded implementation proved to be efficient even without GPU acceleration, demonstrating the feasibility of low-cost AI deployment for infrastructure monitoring.

X Advantages of the Proposed System

Automated Road Inspection

The proposed system eliminates the need for manual road inspection by using an AI-based real-time detection mechanism. Traditional road inspection methods require human labour, which is time-consuming, subjective, and inefficient for large-scale infrastructure monitoring. In contrast, the developed system continuously monitors road conditions using a camera and embedded AI model, ensuring consistent and unbiased detection of road damages.

Key Advantages:

- Reduces human intervention
- Enables continuous monitoring
- Provides objective damage assessment
- Improves inspection efficiency

By automating the inspection process, the system significantly reduces operational costs and enhances the accuracy of infrastructure assessment.

Real-Time Damage Detection

One of the major strengths of the system is its ability to detect road damages in real time using an embedded deep learning model. The YOLO-based detection framework processes live video frames and identifies potholes and cracks instantly, allowing immediate visualization and logging of road defects.

Benefits of Real-Time Detection:

- Instant identification of road damages
- Faster decision-making for maintenance
- Continuous frame-by-frame monitoring
- Immediate visual feedback on screen

This real-time capability makes the system highly suitable for deployment in smart transportation and intelligent infrastructure monitoring applications.

Edge Computing Capability

The system is designed using an edge computing architecture where all computations are performed locally on the Raspberry Pi. This removes the dependency on cloud servers and internet connectivity, making the system more reliable in remote and rural areas.

Advantages of Edge Computing:

- Low latency processing
- Offline functionality
- Reduced network bandwidth usage
- Enhanced data security and privacy

Edge deployment ensures uninterrupted operation even in environments with poor internet connectivity, which is a critical requirement for real-world road monitoring systems.

Geo-Tagged Damage Logging

The integration of a GPS module allows the system to attach location coordinates to each detected road defect. This geo-tagging feature enhances the usability of the system for infrastructure management and smart city applications.

Importance of Geo-Tagging:

- Accurate location tracking of road damages
- Mapping of damage-prone areas
- Efficient maintenance planning
- Data-driven decision making

Even though GPS signals were unavailable during indoor testing, the system architecture fully supports geo-tagging for outdoor deployment.

Low-Cost and Scalable Design

The hardware components used in the system, such as Raspberry Pi and camera module, are cost-effective and easily available. This makes the system economically viable for large-scale deployment across cities and highways.

Scalability Features:

- Easy installation on vehicles
- Portable embedded design
- Expandable storage and sensors
- Suitable for smart city integration

The low-cost nature of the system allows government and municipal bodies to adopt it for continuous road infrastructure monitoring.

XI Future Scope and Enhancements

Cloud Dashboard Integration

In future developments, the system can be integrated with a cloud-based dashboard for centralized monitoring of road damages. This will allow authorities to visualize geo-tagged damage data on digital maps.

Possible Enhancements:

- Real-time damage mapping
- Cloud data analytics
- Remote monitoring interface
- Smart city control center integration

This enhancement will transform the system into a fully intelligent infrastructure monitoring platform.

Edge TPU or GPU Acceleration:

To improve processing speed and detection accuracy, hardware accelerators such as Edge TPU or GPU modules can be integrated with the Raspberry Pi.

Benefits of Hardware Acceleration:

- Increased frame processing speed
- Higher detection accuracy
- Reduced inference latency
- Improved real-time performance

This upgrade will make the system suitable for high-speed vehicle-based inspections.

Severity Estimation and Depth Analysis

Future versions of the system can incorporate depth sensing and severity estimation to measure the depth and size of potholes more accurately.

Advanced Features:

- Depth-based pothole analysis
- Severity scoring models
- Predictive maintenance alerts
- Automated repair prioritization

This will enhance the system's analytical capability beyond simple detection.

Outdoor Field Deployment and Testing

Extensive outdoor testing can be conducted to evaluate system performance under real-world conditions such as highways, urban roads, and rural infrastructure.

Field Testing Benefits:

- Accurate GPS geo-tagging validation
- Real-time vehicle-mounted monitoring
- Environmental robustness evaluation
- Large-scale infrastructure assessment

XII CONCLUSION

The proposed AI-based real-time road damage detection and geo-tagging system successfully demonstrates the effectiveness of embedded edge computing for smart infrastructure monitoring. The integration of a YOLO-based deep learning model with a Raspberry Pi platform enables automated detection of potholes and cracks in real time, eliminating the limitations of traditional manual inspection methods.

The system was experimentally validated through prototype testing, where multiple road damages were accurately detected, classified, and stored with timestamps and priority levels. The console outputs confirmed stable real-time performance, automated image logging, and successful multi-class detection. Although GPS signals were unavailable during indoor testing, the geo-tagging module is fully capable of providing accurate location data in outdoor environments.

Key contributions of the proposed system include:

- Real-time AI-based road damage detection
- Automated evidence storage and logging
- Priority-based damage classification
- Edge computing deployment without cloud dependency
- Cost-effective and scalable embedded design

Overall, the developed system provides a reliable, portable, and intelligent solution for modern road infrastructure monitoring. With future enhancements such as cloud integration, hardware acceleration, and large-scale field deployment, the system has strong potential for smart city applications and predictive road maintenance planning.